REMARKS

In the Office Action, the Examiner noted that claims 1 and 21-39 are pending in the application and that claims 1 and 21-39 are rejected. By this response, claims 1 and 21-39 continue without amendment. In view of the following discussion, Applicants respectfully submit that none of such claims are obvious under the provisions of 35 U.S.C. §103. Thus, Applicants believe that all of these claims are now in condition for allowance.

Rejection of Claims Under 35 U.S.C. §103

The Examiner rejected claims 1 and 21-39 as being unpatentable over Bowen (U.S. Published Patent Application 2003/0105620, published June 5, 2003) in view of Smith et al., "An Architecture Design and Assessment System for Software/Hardware Codesign," IEEE, 1985, pp.417-424 ("Smith"). The rejection is respectfully traversed.

More specifically, the Examiner conceded that Bowen does not teach a step of invoking results of the traversing step using a high-level synthesis compiler to generate a functionality graph and a resource graph for the target architecture. (Office Action, p. 3). The Examiner stated, however, that Smith teaches production of a functionality graph and a resource graph. (Office Action, p. 3). The Examiner concluded that it would have been obvious to modify the hardware architecture program development of Bowen with the functionality and resource graphs taught by Smith. (Office Action, p. 3). Applicants respectfully disagree.

Bowen generally teaches writing first computer code in a first programming language (e.g., Handel-C), where the first computer code references second computer code in a second programming language (e.g., EDIF, VHDL). The second computer code is simulated for use during the execution of the first computer code in the first programming language. (See Bowen, Abstract; Claims 1-20). The Examiner specifically cites paragraph 0145 of Bowen, which generally describes the use of Handel-C to program FPGAs. The Examiner also cites paragraph 0268 of Bowen, which states that a compiler processes Handel-C code to produce a file, which in turn is compiled into native PC code using Microsoft Visual C++.

Smith generally teaches a CAD system that supports the codesign of hardware and software architectures for digital signal processors based on directed graph methodology. (Smith, Abstract). The CAD system employs a software graph to represent the software portion of the design and a hardware graph to represent the hardware portion of the design. (Smith, p. 421-423). The graphs are created by a designer and analyzed by the CAD system to assess performance. (Smith, p. 419-420).

The cited references, either singly or in any permissible combination, do not teach, suggest, or otherwise render obvious Applicants' invention recited in claim 1. Namely, the combination of Bowen and Smith does not teach or suggest invoking results of traversing an intermediate graph representation using a high-level synthesis compiler to generate a functionality graph and resource graphs for the target architecture. Compiling a file with Microsoft Visual C++, as taught by Bowen does not teach or suggest both traversing an intermediate representation of the RDL description and invoking the results of the traversal using a compiler to generate graphs for the target architecture. At best, compiling a file with Microsoft Visual C++ might arguably teach traversing a file. There is, however, no teaching or suggestion in Bowen that the Microsoft Visual C++ compiler generates graphs for the target architecture. Nor is there any teaching or suggestion in Bowen of invoking the results of the Microsoft Visual C++ compiler using a high-level synthesis compiler.

Smith generally describes use of directed graphs to represent software and hardware portions of a design. Applicants', however, do not merely claim the use of directed graphs. Rather, Applicants' specifically recite a step of invoking results of traversal using a high-level synthesis compiler to generate a functionality graph and resource graphs. The context in which the functionality graph and resource graphs are generated must be considered. There is no teaching or suggestion in Smith of invoking results using a high-level synthesis compiler to generate the software/hardware graphs. Rather, the graphs in Smith are generated by the designer and then analyzed by the tools. (Smith, p. 424, section entitled "Graphics Editor and Viewer") (stating that [a] user adds a node to the graph by selecting from a menu of templates and adding or modifying the appropriate attributes."). Since neither Bowen

nor Smith teach or suggest invoking results using a high-level synthesis compiler, the final step in Applicants' claim is not taught or suggested by the combination of references.

Moreover, it is not clear how the generation of hardware/software graphs, as taught by Smith, would be used in the system of Bowen. In Bowen, a compiler processes Handel-C code to produce a file, which in turn is compiled into native PC code using Microsoft Visual C++. The PC code in Bowen (i.e., the results of traversal) is not invoked by a high-level synthesis compiler to produce any type of graphs. The graphs cannot be input to the Microsoft Visual C++ compiler for any type of processing, as they are not C++ source files. Rather, Bowen and Smith merely describe two different flows of designing a system. In Bowen, a designer uses Handel C, and in Smith a designer produces hardware/software graphs. The two design flows do not mesh to teach or suggest Applicants' invention recited in claim 1.

Claims 21-39 depend, either directly or indirectly, from claim 1 and recite additional features therefor. Since the cited references do not render obvious Applicants' invention as recited in claim 1, dependent claims 21-39 are also nonobvious and are allowable. Accordingly, Applicants contend that claims 1 and 21-39 are nonobvious over the combination of Bowen and Smith and, as such, fully satisfy the requirements of 35 U.S.C. §103. Applicants respectfully request that the present rejection of such claims be withdrawn.

CONCLUSION

Thus, Applicants submit that none of such claims are obvious under the provisions of 35 U.S.C. §103. Consequently, Applicants believe that all these claims are presently in condition for allowance. Accordingly, both reconsideration of this application and its swift passage to issue are earnestly solicited.

If, however, the Examiner believes that there are any unresolved issues requiring the maintenance of any adverse final action in any of the claims now pending in the application, it is requested that the Examiner telephone Justin Liu at (408) 879-4641 so appropriate arrangements can be made for resolving such issues as expeditiously as possible.

All claims should be now be in condition for allowance and a Notice of Allowance is respectfully requested.

Respectfully submitted,

Justin Liu
Attorney for Applicants
Reg. No. 51,959

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450, on July 19, 2007.

Katherine Stofer
Name

Signature